Assignment 4

1. What’s the difference between final, finally? What is finalize()?

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Key | final | finally | finalize |
| 1. | Definition | final is the keyword and access modifier which is used to apply restrictions on a class, method or variable. | finally is the block in Java Exception Handling to execute the important code whether the exception occurs or not. | finalize is the method in Java which is used to perform clean up processing just before object is garbage collected. |
| 2. | Applicable to | Final keyword is used with the classes, methods and variables. | Finally block is always related to the try and catch block in exception handling. | finalize() method is used with the objects. |
| 3. | Functionality | (1) Once declared, final variable becomes constant and cannot be modified.  (2) final method cannot be overridden by sub class.  (3) final class cannot be inherited. | (1) finally block runs the important code even if exception occurs or not.  (2) finally block cleans up all the resources used in try block | finalize method performs the cleaning activities with respect to the object before its destruction. |
| 4. | Execution | Final method is executed only when we call it. | Finally block is executed as soon as the try-catch block is executed.  It's execution is not dependant on the exception. | finalize method is executed just before the object is destroyed. |

1. What’s the difference between throw and throws?

The Java **throw** keyword is used to throw an exception explicitly.

We specify the exception object which is to be thrown. The Exception has some message with it that provides the error description. These exceptions may be related to user inputs, server, etc.

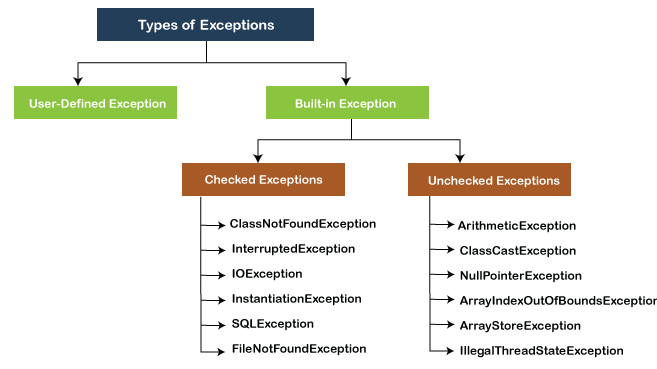
We can throw either checked or unchecked exceptions in Java by throw keyword. It is mainly used to throw a custom exception. We will discuss custom exceptions later in this section.

We can also define our own set of conditions and throw an exception explicitly using throw keyword. For example, we can throw ArithmeticException if we divide a number by another number. Here, we just need to set the condition and throw exception using throw keyword.

The Java **throws** keyword is used to declare an exception. It gives an information to the programmer that there may occur an exception. So, it is better for the programmer to provide the exception handling code so that the normal flow of the program can be maintained.

Exception Handling is mainly used to handle the checked exceptions. If there occurs any unchecked exception such as NullPointerException, it is programmers' fault that he is not checking the code before it being used.

1. What are the two types of exceptions?



1. What is error in java?

In Java, an error is a subclass of Throwable that tells that something serious problem is existing and a reasonable Java application should not try to catch that error. Generally, it has been noticed that most of the occurring errors are abnormal conditions and cannot be resolved by normal conditions. As these errors are abnormal conditions and should not occur, thus, error and its subclass are referred to as Unchecked Exceptions. In Java, we have the concept of errors as well as exceptions. Thus there exist several differences between an exception and an error. Errors cannot be solved by any handling techniques, whereas we can solve exceptions using some logic implementations. So, when an error occurs, it causes termination of the program as errors cannot try to catch.

1. Exception is object, true or false?

True

1. Can a finally block exist with a try block but without a catch?

Yes

1. From java 1.7, give an example of the try-resource feature.

The try-with-resources statement is a try statement that declares one or more resources. A resource is an object that must be closed after the program is finished with it. The try-with-resources statement ensures that each resource is closed at the end of the statement.

public static void writeToFileZipFileContents(String zipFileName,

String outputFileName)

throws java.io.IOException {

java.nio.charset.Charset charset =

java.nio.charset.StandardCharsets.US\_ASCII;

java.nio.file.Path outputFilePath =

java.nio.file.Paths.get(outputFileName);

// Open zip file and create output file with

// try-with-resources statement

try (

java.util.zip.ZipFile zf =

new java.util.zip.ZipFile(zipFileName);

java.io.BufferedWriter writer =

java.nio.file.Files.newBufferedWriter(outputFilePath, charset)

) {

// Enumerate each entry

for (java.util.Enumeration entries =

zf.entries(); entries.hasMoreElements();) {

// Get the entry name and write it to the output file

String newLine = System.getProperty("line.separator");

String zipEntryName =

((java.util.zip.ZipEntry)entries.nextElement()).getName() +

newLine;

writer.write(zipEntryName, 0, zipEntryName.length());

}

}

}

1. What will happen to the Exception object after exception handling?

The Exception object will be garbage collected in the next garbage collection.

1. Can we use String as a condition in switch(str){} clause?

Yes

1. What’s the difference between ArrayList, LinkedList and vector?

图示

描述已自动生成

from the hierarchy diagram, they all implement list interface. they are very similar to use. their main difference is their implementation which causes different performance for different operations. ArrayList is implemented as a resizable array. as more elements are added to ArrayList, its size is increased dynamically. it's elements can be accessed directly by using the get and set methods, since ArrayList is essentially an array. LinkedList is implemented as a double linked list. its performance on add and remove is better than ArrayList, but worse on get and set methods. vector is similar with ArrayList, but it is synchronized. ArrayList is a better choice if your program is thread-safe. vector and ArrayList require space as more elements are added. vector each time doubles its array size, while ArrayList grow 50% of its size each time. LinkedList, however, also implements queue interface which adds more methods than ArrayList and vector, such as offer(), peek(), poll(), etc. note: the default initial capacity of an ArrayList is pretty small. it is a good habit to construct the ArrayList with a higher initial capacity. this can avoid the resizing cost.

1. What’s the difference between HashTable and HashMap?

| S. No. | Hashmap | Hashtable |
| --- | --- | --- |
| 1. | No method is synchronized. | Every method is synchronized. |
| 2. | Multiple threads can operate simultaneously and hence hashmap’s object is not thread-safe. | At a time only one thread is allowed to operate the Hashtable’s object. Hence it is thread-safe. |
| 3. | Threads are not required to wait and hence relatively performance is high. | It increases the waiting time of the thread and hence performance is low. |
| 4. | Null is allowed for both key and value. | Null is not allowed for both key and value. Otherwise, we will get a null pointer exception. |
| 5. | It is introduced in the 1.2 version. | It is introduced in the 1.0 version. |
| 6. | It is non-legacy. | It is a legacy. |

1. What is static import?

In Java, static import concept is introduced in 1.5 version. With the help of static import, we can access the static members of a class directly without class name or any object. For Example: we always use sqrt() method of Math class by using Math class i.e. Math.sqrt(), but by using static import we can access sqrt() method directly.

According to SUN microSystem, it will improve the code readability and enhance coding. But according to the programming experts, it will lead to confusion and not good for programming. If there is no specific requirement then we should not go for static import.

1. What is static block?

In simpler language whenever we use a static keyword and associate it to a block then that block is referred to as a static block. Unlike C++, Java supports a special block, called a static block (also called static clause) that can be used for static initialization of a class. This code inside the static block is executed only once: the first time the class is loaded into memory.

1. Explainthe keywords:

default(java 1.8), break, continue, synchronized, strictfp, transient, volatile, instanceOf

* **default** The default keyword specifies some code to run if there is no case match
* **break** When a break statement is encountered inside a loop, the loop is immediately terminated and the program control resumes at the next statement following the loop. The Java break statement is used to break loop or switch statement. It breaks the current flow of the program at specified condition. In case of inner loop, it breaks only inner loop.We can use Java break statement in all types of loops such as for loop, while loop and do-while loop.
* **continue** The continue statement is used in loop control structure when you need to jump to the next iteration of the loop immediately. It can be used with for loop or while loop. The Java continue statement is used to continue the loop. It continues the current flow of the program and skips the remaining code at the specified condition. In case of an inner loop, it continues the inner loop only. We can use Java continue statement in all types of loops such as for loop, while loop and do-while loop.
* **synchronized** Methods can only be accessed by one thread at a time
* **strictfp** strictfp is a keyword in java used for restricting floating-point calculations and ensuring same result on every platform while performing operations in the floating-point variable.
* **transient** Attributes and methods are skipped when serializing the object containing them
* **volatile** The value of an attribute is not cached thread-locally, and is always read from the "main memory"
* **instanceof** The instanceof keyword checks whether an object is an instance of a specific class or an interface. The instanceof keyword compares the instance with type. The return value is either true or false.

1. Create a program including two threads – thread read and thread write.

Input file -> Thread read -> Calculate -> buffered area

Buffered area -> Thread write -> output file

Detailed description is in assignment4.txt file.

Sample input.txt file.

Attached files are input.txt and a more detailed description file.

文本

描述已自动生成

文本

描述已自动生成